

TelePort Modules

- HTTP
- Poll
- Socket



State Variables

- __RPC_SUCCESS__
- __RPC_ERROR__
- __RPC_TIMEOUT__

Extra Object

- id
- userdata
- http
- url
- tpModule
- message
- Retries

Attributes

RPC

- id
- protocol
- url
- url-eval
- method-name
- autoroute
- multicall
- ns-name
- ns-url
- offline

Method

- name
- receive
- async
- export
- type
- variable
- lookup
- caching

RPC Classes

- POST
- HEADER
- REST
- SOAP
- XMLRPC
- JSON
- JPHP

RPC Properties

- isRPC
- useHTTP
- routeServer
- autoroute
- namedArguments
- protocol
- multicall
- useXML

RPC Methods

- addMethod
- setCallback
- revert
- purge
- load
- serialize
- unserialize

HTTP Properties

- timeout

HTTP Methods

- toString
- saveCache
- loadCache
- getXML
- get
- cancel
- retry

Javeline IFDEF's

- __WITH_TELEPORT
- __TP_IFRAME
- __TP_SOCKET
- __TP_POLL
- __TP_RPC
- __TP_RPC_HEADER
- __TP_RPC_JPHP
- __TP_RPC_JSON
- __TP_RPC_POST
- __TP_RPC_REST
- __TP_RPC_SOAP
- __TP_RPC_XMLRPC

POST Example

Javeline Markup Language:

```
<RPC id="comm" protocol="POST" url-eval="HOST_PATH+'/'subscribe.cgi" method-name="func" autoroute="true">
  <method name="login">
    <variable name="username" />
    <variable name="password" />
    <variable name="status" />
  </method>
</RPC>
```

Javascript:

```
<script>
  comm.login("ruben", "passwd", "online");
</script>
```

XML-RPC Example (advanced options)

Javeline Markup Language:

```
<RPC id="comm" protocol="XMLRPC" url="subscribe.cgi">
  <global name="version" value="1.0" />
  <method name="login" type="global" variable="hash" />
  <method name="getUsers">
  <method name="saveUser">
  <method name="removeUser">
  <method name="getStates">
  <method name="fillstates">
  <method name="dolt" async="false">
</RPC>
```

SOAP Example

Javeline Markup Language:

```
<RPC id="Amazon" protocol="SOAP" url="http://webservices.amazon.com/onca/soap?Service=AWSECommerceService" receive="rcvBooks">
  <method name="ItemSearch">
    <variable name="AWSAccessKeyId" value="fixed_value"/>
    <variable name="MerchantId" default="my_default_value"/>
    <variable name="Condition" />
    <variable name="Availability" />
    <variable name="SearchIndex" />
    <variable name="Keywords" encoded="true"/>
  </method>
</RPC>
```

Callback Example

```
function rcvBooks(xmlData, state, extra){
  if(state != __RPC_SUCCESS__){
    if(state == __HTTP_TIMEOUT__ && extra.retries < MAX_RETRIES)
      return extra.tpModule.retry(extra.id);
    else throw new Error(0, "Error");
  }
  alert(xmlData.xml);
}
comm.setCallback("ItemSearch", rcvBooks);
```

Model Initialization

Model tag

```
<j:Model file="" />
<j:Model rpc="" arguments="" />
```

Model attribute

```
model="file:http://asasda.com"
model="rpc:obj.call:arg1:arg2:arg3"
```

Arguments

```
Literal:      someName
XPath:       xpath:/path/to/node
eval:        eval:5+10
method:      method:mName
array:       ()
```

example: 10;(xpath:text();eval.5+10)

SmartBinding Rules

Javeline Markup Language:

```
<j:smartbinding>
  <j:bindings>
    <Load select="." rpc="" arguments="" />
    <Insert select="." rpc="" arguments="" />
  </j:bindings>
  <j:actions>
    <Rename select="." rpc="" arguments="" />
    <Remove select="." file="removeUser.php" />
  </j:actions>
</j:smartbinding>
```